PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 4 : G07F 7/10, H04L 9/00 — A1

(11) International Publication Number: WO 89/11706

(43) International Publication Date: 30 November 1989 (30.11.89)

(21) International Application Number: PCT/US89/01944

(22) International Filing Date: 4 May 1989 (04.05.89)

(30) Priority data:

8811816.1	19 May 1988 (19.05.88)	GB
8906496.8	21 March 1989 (21.03.89)	GB
331,788	3 April 1989 (03.04.89)	US

(81) Designated States: AU, CH (European patent), DE (European patent), FR (European patent), GB (European patent), JP, NL (European patent).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(71) Applicant: NCR CORPORATION [US/US]; World Headquarters, Dayton, OH 45479 (US).

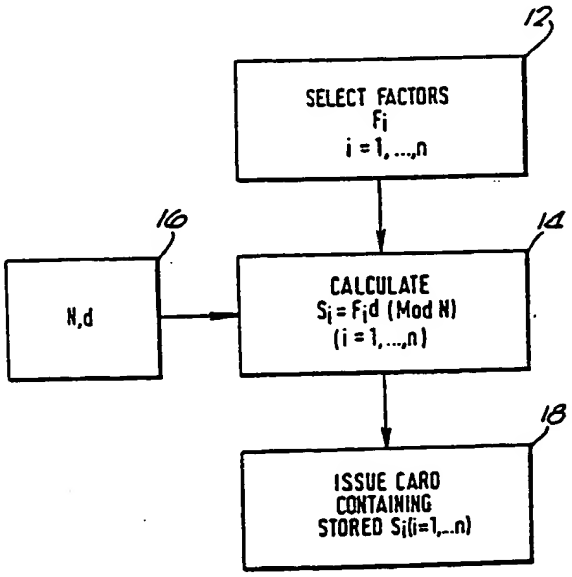
(72) Inventor: AUSTIN, Jeffrey, Reginald ; The White House, Tilford Road, Hindhead, Surrey GU26 6TD (GB).

(74) Agents: JEWETT, Stephen, F. et al.; Patent Division, NCR Corporation, World Headquarters, Dayton, OH 45479 (US).

(54) Title: METHOD AND DEVICE FOR AUTHENTICATION

(57) Abstract

An entity such as a smart card (30) includes microprocessor means (36), input/output means (44) and PROM storage means (42) which stores a set of transformations S_i (i = 1, ..., n) of a corresponding set of public factors F_i (i = 1, ..., n), where $S_i = F_i^d \pmod N$, d being the secret key counterpart of a public key e associated with the modulus N, which is the product of two primes. An authentication device (32) which stores the public factors F_i and the values of N and e, generates an n-bit random vector $V = v_i$ which is transmitted to the card (30) where a product Y of the values S_i selected according to the 1-bits of V is computed and transmitted to the authentication device (32) which computes $X_{act} = Y^e \pmod N$ and also computes X_{ref}, the product of the F_i selected according to the 1-bits of V. If X_{act} and X_{ref} are equal, then the card is authenticated to within a certain probability. An analogous method is disclosed for certifying messages to be transmitted. In further embodiments, a higher degree of security is achieved by arranging for the entity being authenticated, or the certifying entity, to select an additional secret factor or plurality of secret factors.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	FI	Finland	ML	Mali
AU	Australia	FR	France	MR	Mauritania
BB	Barbados	GA	Gabon	MW	Malawi
BE	Belgium	GB	United Kingdom	NL	Netherlands
BF	Burkina Faso	HU	Hungary	NO	Norway
BG	Bulgaria	IT	Italy	RO	Romania
BJ	Benin	JP	Japan	SD	Sudan
BR	Brazil	KP	Democratic People's Republic of Korea	SE	Sweden
CF	Central African Republic			SN	Senegal
CG	Congo	KR	Republic of Korea	SU	Soviet Union
CH	Switzerland	LI	Liechtenstein	TD	Chad
CM	Cameroon	LK	Sri Lanka	TG	Togo
DE	Germany, Federal Republic of	LU	Luxembourg	US	United States of America
DK	Denmark	MC	Monaco		
ES	Spain	MG	Madagascar		

METHOD AND DEVICE FOR AUTHENTICATION

This invention relates to the authentication of devices and messages.

It is a common requirement to verify the authenticity of data which may represent monetary value or may imply the authenticity of the entity generating that data.

To impede forgery, only a manufacturing source which produces entities should possess the means to produce authentication devices for the entities. This implies that the source must possess some secret. The difficulty in proving authenticity is in providing the means to the authenticator to achieve that proof. Many systems employ an algorithm driven by a secret key such that a data string passed through the algorithm results in a secret transformation of that data. The data so transformed is used as an authentication certificate or code which may be tested by an authenticator. One method of testing involves the authenticator in performing the same secret transformation of the data to yield an authentication certificate which is compared for equality with that provided by the source entity.

The problem with this technique is that the authenticator must duplicate the data manipulation by the source so as to compare the result for equality. This means that an authenticator can forge an authentication certificate and claim that it emanated from the source. Another problem is that the authenticator must also have knowledge of the key. This problem is particularly acute if several entities need to authenticate another entity, since each must possess the secret key. Disclosure of this key by one authenticator therefore compromises all authenticators and the source. Furthermore, the secret key must be securely distributed

to each potential authenticator prior to the event. This therefore limits the ability to authenticate to only those trusted entities which were anticipated to require the function.

Where it may be necessary for a large number of unpredictable entities to possess the ability to authenticate another entity, the use of secret key algorithms is somewhat impractical. Further, when it is desirable that the authenticator be completely denied the ability to forge an authentication certificate the duplicative equality test method cannot be employed.

Another known technique employs the art of public key cryptography wherein an asymmetrical algorithm is used. Public key cryptography is described in the article: Communications of the ACM, vol. 21, No. 2, February 1978, pages 120-126, R.L. Rivest et al. " A Method for Obtaining Digital Signatures and Public Key Cryptosystems". In this known technique, a data element or a change sensitive compression of a data string is enciphered using a secret key or procedure. Authenticity is proven by obtaining the original data element (or change sensitive compression) which is used as a reference value and then using a public key or procedure to decipher the data supplied by the source. Equality of the deciphered data with the reference data implies that the secret key or procedure was employed and thus that the data is authentic.

This technique permits any entity to know the public key or procedure with which to prove the authenticity of data emanating from an entity possessing the complementary secret key or procedure. Consequently, the key distribution problem is significantly eased as prior knowledge and secrecy are not required.

However, the publicly known procedure must not permit the secret key or procedure to be easily determined. Generally, the algorithms possessing this property require substantial computing power to perform the secret procedure. This usually renders them unsuitable for low cost devices where operational speed is a requirement. If multiple portable devices or the data emanating from them must be able to be tested for authenticity, then the secret key and algorithm must be contained in each device. In this case, disclosure of the secret key in one device will compromise all similar devices.

This technique is therefore not practical for low cost replicated devices.

European Patent Application No. 0 252 499 discloses a method for creating a unique card identifier in the form of a "smart card" which involves selecting a modulus which is a product of two primes, preparing a string of information unique to the card identifier, utilizing a pseudo-random function to transform such string and a plurality of selected indices to derive an associated plurality of values which are quadratic residues with respect to the modulus, computing the square roots of the reciprocals of the quadratic residues, and recording the information string, such square roots and the related indices in the card identifier. Such card is authenticated by transmitting the information string and the selected indices from the card to a verification device and generating in the verification device the quadratic residues utilizing the pseudo-random function, selecting in the card a random number, computing the squared value of the random number and transmitting such squared value from the card to the verification device, generating in the verification device a random vector which is sent to the card, computing in the card the product of the random number and a selection of the

stored square root values dependent on the random vector, transmitting the product to the verification device, squaring the transmitted product and multiplying such squared value by a selection of the computed quadratic residue values selected in accordance with the random vector, and checking that the result value is equal to the squared random number. This known method is complex and in particular involves the selection and utilization of quadratic residue values.

It is an object of the present invention to provide a relatively simple method and apparatus for the authentication of devices and messages.

Therefore, according to a first aspect of the present invention, there is provided a method of manufacturing an entity, including the steps of:

(a) selecting a modulus N which is a product of at least two prime numbers;

(b) selecting an integer e which is relatively prime to $\varphi(N)$, where $\varphi(N)$ is Euler's totient function of N; and

(c) determining an integer d such that $e.d = 1 \pmod{\varphi(N)}$, characterized by the steps of:

(d) selecting a set of n public factors $F_1, \ldots, F_n$ $(0<F_i<N)$;

(e) calculating $S_i = F_i^d \pmod{N}$ for $i=1, \ldots, n$; and

(f) storing the n values S_i $(i=1, \ldots, n)$ and the value N in said entity.

According to a second aspect of the invention, there is provided a method of authenticating an entity according to the first aspect of the invention, characterized by the steps of:

(j) placing said entity in communication with an authentication device;

(k) generating in said authentication device an n-bit

binary string $V = v_i$ (i=1, ..., n)

(l) transmitting said binary string V to said entity;

(m) calculating, in said entity

$$Y = \prod_{v_i=1} S_i \pmod N;$$

(n) transmitting Y to said authentication device;

(o) calculating, in said authentication device

$$X_{ref} = \prod_{v_i=1} F_i \pmod N; \text{ and}$$

$$X_{act} = Y^e \pmod N; \text{ and}$$

(p) comparing X_{ref} and X_{act}.

According to a third aspect of the invention, there is provided a method of certifying a message M generated by or presented to an entity manufactured according to the first aspect of the invention, characterized by the steps of:

(q) computing a change-sensitive transformation H of said message M;

(r) generating an n-bit binary string $V = v_i$ (i=1, ..., n), using the computed value of H;

(s) computing

$$Y = \prod_{v_i=1} S_i \pmod N; \text{ and}$$

(t) appending Y as a message authentication code (MAC) certificate to said message M.

According to a fourth aspect of the invention, there is provided an entity including processing means, input/output means and memory means, characterized in that said memory means has stored therein a modulus N which is the product of at least two prime numbers and a set of n factors S_i (i=1, ..., n) where

$$S_i = F_i^{-d} \pmod N,$$

where d is the secret key counterpart of a public key e, associated with the modulus N, and F_i (i=1, ..., n) are n public factors, $0<F_i<N$, and wherein said processing

means is adapted to compute

$$Y = \prod_{v_i=1} S_i \pmod N$$

where $V = v_i$ is an n-bit binary string.

According to a fifth aspect of the invention, there is provided an authentication device for use with an entity according to the fourth aspect of the invention, including further processing means, further input/output means and further memory means, characterized in that said further memory means has stored therein said n public factors F_i (i=1, ..., n), said modulus N, and said public key e, and wherein said further processing means is adapted to compute

$$X_{ref} = \prod_{v_i=1} F_i \pmod N; \text{ and}$$

$$X_{act} = Y^e \pmod N$$

using the stored values of F_i, N and e, and to compare X_{ref} with X_{act}.

Embodiments of the present invention will now be described by way of example, with reference to the accompanying drawings, in which:-

Fig. 1 is a block diagram showing the procedure utilized by a card issuer in creating a smart card;

Fig. 2 is a block diagram of a card in operative association with a card acceptor device;

Fig. 3 is a block diagram of a message source unit;

Fig. 4 is a block diagram of a message authentication unit; and

Fig. 5 is a diagram showing the map of a memory utilized in an alternative embodiment of the invention.

Firstly, the theoretical basis underlying the invention will be explained, as an aid to understanding the invention. It is known that, if N is the product of (at least) two prime numbers P, Q, i.e., if

$$N = P.Q;$$

and if e is relatively prime to $\varphi(N)$, where

$$\varphi(N) = (P-1).(Q-1)$$

is Euler's totient function (the number of integers less than N which are relatively prime to N), then, in modulus N arithmetic, a value d can be determined (see for example, the aforementioned article by Rivest et al) which is the multiplicative inverse of e such that

$$e.d = 1 \pmod{\varphi(N)}.$$

The value d is commonly referred to as the secret key counterpart of the public key e.

Thus, if

$$X = Y^e \pmod N,$$

then

$$Y = X^d \pmod N$$

for all values of Y, $0<Y<N$.

Furthermore, if

$$X = F_1 . F_2 .... F_n \pmod N \qquad (1)$$

where F_i ($i = 1, \ldots, n$) are integer values, with

$$0<F_i<N$$

then

$$X^d = F_1^d . F_2^d .... F_n^d \pmod N$$

and

$$X^d \pmod N = \{F_1^d \pmod N . F_2^d \pmod N \ldots F_n^d \pmod N\} \pmod N$$

Let

$$S_i = F_i^d \pmod N;\ i=1, \ldots, n \qquad (2)$$

Then

$$X^d \pmod N = S_1 . S_2 \ldots S_n \pmod N$$

Let

$$Y = X^d \pmod N$$

Therefore

$$Y = S_1 \cdot S_2 \ldots S_n \pmod{N} \quad (3)$$

Let V represent a binary string of n bits, $V = v_1 \ldots v_n$ such that each bit v_i of V is a flag indicating the inclusion of the corresponding $F_1, \ldots, F_n$ and $S_1, \ldots, S_n$ in the calculation of X and Y respectively, so that

$$X = \prod_{v_i=1} F_i \pmod{N}. \quad (4)$$

From (3)

$$Y = \prod_{v_i=1} S_i \pmod{N} \quad (5)$$

Therefore, provided that the N and d values employed in (1) and (2) satisfy the above requirements, then

$$X = \prod_{v_i=1} F_i \pmod{N} = \{\prod_{v_i=1} S_i \pmod{N}\}^e \pmod{N}$$

$$= Y^e \pmod{N}$$

for all values of F_i, $0<F_i<N$.

With the above in mind, a first embodiment of the invention will now be described, wherein multiple low cost devices, in the form of entities which will be referred to in the descriptions of the preferred embodiments as smart cards, are produced by a card issuer and distributed to individuals. The embodiment enables such issued cards to be expeditiously authenticated by verifying devices.

Referring first to Fig 1, a card issuer selects, as shown at box 12, a plurality of n public factors F_i ($i=1, \ldots, n$), where $0<F_i<N$, and such factors, together with the value of the modulus N and the value of e are made publicly available to authenticators, that is, organizations which may wish to authenticate smart cards issued by the smart card issuer. In a particular application a suitable value for n is 32, and the value of N is in the range $2^{512}<N<2^{513}$.

The card issuer computes the n values S_i, where

$S_i = F_i^d \pmod N \quad i=1, \ldots, n$

as shown at box 14, using provided values of N and d (box 16), where d is maintained secret. These values S_i are also maintained secret. The card issuer then issues cards which contain n values S_i (i=1, ..., n) stored in a secure manner, for instance in a secure PROM. It should be understood that by a "secure PROM" herein is meant a PROM the contents of which are protected from unauthorized read-out, for example, such protection may involve software protection and hardware protection in the form of shielding.

When it is desired to authenticate a smart card 30, Fig. 2, the card 30 is inserted into a card acceptor device 32, whereby a data communication path 34 is established between the smart card 30 and the card acceptor device 32.

The smart card 30 includes a microprocessor 36, a RAM 38, a program PROM 40 which stores the program controlling the operation of the card 30, a secure PROM 42 containing the n values S_i (i=1, ..., n) stored in respective storage locations 102-1 to 102-n and the value N stored in a storage location 104, and an input/output unit 44. Alternatively, since N is a public value, it could be stored in the RAM 38. The devices 36, 38, 40, 42 and 44 within the card are interconnected by a communications bus 46.

The card acceptor device 32 includes a microprocessor 50, a RAM 52, a program PROM 54 which stores the program controlling the operation of the acceptor device 32, a keyboard 56, a display 58, a printer 60, a random number generator 62, and an input/output unit 64. The RAM 52 includes storage locations 112-1 to 112-n storing the n public factors $F_1, \ldots, F_n$ and storage locations 114, 116 storing the values N and e, respectively. The various units located in the card acceptor device 32 are

interconnected by a communications bus 66.

When a card 30 inserted into the card acceptor device 32 is to be checked for authenticity, the random number generator 62 generates an n-bit random number V having n bits v_i (i=1, ..., n). In order to ensure that V contains at least two bits equal to binary 1, the microprocessor 50 is controlled, if necessary, to set the least significant bits of V progressively to binary 1 until at least two binary 1 bits are present in V. Thus, if the initial value of V is all zero bits, then the two least significant bits are set to binary 1. The value V is stored in the RAM 52.

The value V is then transmitted from the RAM 52 via the input/output unit 64 over the communication path 34 and the input/output unit 44 and is stored in the RAM 38 contained in the card 30. The microprocessor 36 checks that V contains at least two binary 1 bits, and if so, computes the value Y where

$$Y = \prod_{v_i=1} S_i \pmod N$$

using the values S_i stored in the PROM 42.

The value Y is then transmitted via the input/output unit 44, the transmission path 34 and the input/output unit 64 and is stored in the RAM 52. Using the values F_i (i=1, ..., n) V, and e, stored in the RAM 52, the microprocessor 50 then computes

$$X_{ref} = \prod_{v_i=1} F_i \pmod N$$

and

$$X_{act} = Y^e \pmod N,$$

and tests whether

$$X_{ref} = X_{act}.$$

Equality implies the authenticity of the X_{act} response

with probability of 1:N. The authenticity of the card 30 producing the response has a probability of $1:2^n-n$. By issuing repetitive random challenges in the form of random values of V, the probability that the card 30 is authentic increases exponentially by $1:(2^n-n)^j$ where j is the number of challenges issued.

It will be appreciated that the card 30 needs only to compute

$$Y = \prod_{v_i=1} S_i \pmod N$$

to respond to a challenge. Since this is at most n-1 multiplications using modulo N arithmetic, the work factor is significantly less than $Y = X_{ref}^{d} \pmod N$ for any large value of d. In this connection, it will be appreciated that since d is in effect the secret key associated with the card 30, and given that

$$e.d = 1 \pmod{\varphi(N)}$$

then d will be in the order of magnitude of 2N/3 for convenient values of e. Thus, in the described embodiment, authentication security comparable to that achievable with public key digital signature methods is achieved with significantly less computational effort. Furthermore, with no secret key used during the authentication process, it is possible to produce multiple cards 30 loaded with the $S_1, \ldots, S_n$ values which may be dynamically challenged by a verifying device to achieve similar confidence levels to those obtained with public key digital signature authentication methods.

It will be appreciated that the result of the authentication procedure can be indicated on the display 58 and/or recorded by the printer 60.

In a second embodiment of the invention, a data string forming a message M is authenticated by appending a certificate thereto. Such message M could, for example, be a data string representing a legal document, a

program file, or other information. Referring to Fig. 3, there is shown a message source unit 30A, which includes a message buffer 70 adapted to temporarily store a message M to be authenticated. The message source unit 30A further includes a microprocessor 36A, a RAM 38A, a program PROM 40A, a secure PROM 42A and an input/output unit 44A connected to a communications path 34A. The message source unit 30A also includes a communications bus 46A interconnecting devices 36A, 38A, 40A, 42A, 44A and 70 therein. It will be appreciated that the devices having the references with suffix A in Fig. 3 correspond to similarly referenced devices in the smart card 30 shown in Fig. 2, and in a practial implementation, the message source unit 30A could be a smart card. Furthermore, the secure PROM 42A stores the values S_1, S_2, ... S_n in locations 102A-1 to 102A-n, the value of the modulus N in storage location 104A and the value of e in storage location 106A. Clearly, the values of N and e, being public values, could alternatively be stored in the RAM 38a.

A message M stored in the message buffer 70 is authenticated by appending thereto a message authentication code (MAC) which is computed in the following manner.

Using the stored values of N and e, the microprocessor 36A first computes a change-sensitive transformation H of the message M. In the preferred embodiment, this is effected by computing:

$$H = M^e \pmod N$$

The value H is then converted to a binary value J, which is segmented into sub-fields of length n (with padding of an incomplete field with predetermined binary bits if necessary) and the individual sub-fields are added together modulo 2 (exclusive-or operation) such that the resultant binary string is used as $V = v_i$ (i=1, ..., n) in the calculation of Y, where

$$Y = \prod_{v_i=1} S_i \pmod N,$$

as described in the first embodiment.

This value of Y is then appended as a message authentication code (MAC) when the message M is transmitted from the message source unit 30A via the input/output unit 44A to a communication path 34A.

An authentication device 32A, Fig. 4, which is of generally similar construction to the card acceptor device 32 shown in Fig. 2 may be used to authenticate the transmitted message M. The authentication device 32A includes a message buffer 72, a RAM 52A, a program PROM 54A, a keyboard 56A, a display 58A, a printer 60A, an input/output unit 64A and an interconnecting communications bus 66A.

Stored in the RAM 52A, in locations 112A-1 to 112A-n, 114A and 116A, are the public factor values $F_1, \ldots, F_n$, together with the public key e and modulus N.

The message M, received over the communications path 34A is stored in the message buffer 72, together with the MAC, Y.

Using the received message M, the microprocessor 50A computes H and J to obtain V as in the message source unit 30A, and then computes

$$X_{ref} = \prod_{v_i=1} F_i \pmod N$$

utilizing the public factors F_i stored in the RAM 52A.

Using the received value Y stored in the message buffer 70, the microprocessor 50A then computes

$$X_{act} = Y^e \pmod N.$$

Finally, the values of X_{act} and X_{ref} are compared using the microprocessor 50A. Equality of X_{act} and X_{ref}

implies that the message source unit 30A possessed S_1, ..., S_n, and thus that the message M is authentic, within a probability of 1:N. It will be appreciated that this embodiment has the advantage that a low cost device (message source unit 30A) may readily certify data emanating from it with a probability of 1:N.

It should be understood that in the second embodiment, as in the first embodiment, in order to protect the S_i values from disclosure, it must be ensured that V contains at least two binary 1 bits, by progressively setting the least significant bits of V to binary 1 if necessary.

The second embodiment of the invention has the further advantage that several message source units 30A or the data emanating therefrom may be authenticated without the unit actually being present at the time of authentication. This ability is particularly useful for authenticating messages which may have been produced some time earlier by various message source units 30A, in the form of low cost devices such as smart cards. Multiple message source units may share the same F_1, ..., F_n values which would be standardized for the scheme, with individual integrity being ensured by various values of e and N.

However, it is preferred to standardize e and F_1, ..., F_n for all users of an authentication scheme within a group of users and for the operator of each message source unit to publish a specific value N to be used for his message source unit. Should an operator possess several such units, rather than specifying a unique value of N for each unit, integrity can be assured in a manner which will now be described with reference to the third embodiment of the invention.

According to a third embodiment of the invention, a

message M may be authenticated as originating from a unique message source unit among a set of such message source units sharing the same $F_1, \ldots, F_n$ and N and e values. This has the advantage that it is infeasible for one member of such a set to masquerade as another member of the set. For this purpose, the operator of the system allocates to each message source unit a public factor F_{ID} which is unique to that source unit. Furthermore, the operator of the system computes, for each such F_{ID} value, a corresponding S_{ID} value;

$$S_{ID} = F_{ID}^{d} \pmod{N},$$

where d is the system secret key, and stores S_{ID} in the secure memory of the relevant message source unit.

Referring to Fig. 5, there is shown a diagram of the secure PROM 42B included in the message source unit. The PROM 42B contains storage locations 102B-1 to 102B-n storing the n values $S_1, \ldots, S_n$, respectively, storage locations 104B and 106B storing the values N, e, respectively, and storage locations 108, 110, storing the values F_{ID}, S_{ID}, respectively.

In the third embodiment, it should be understood that the operation is generally similar to that described for the second embodiment, except that the calculation of the MAC, Y, is made according to the formula

$$Y = S_{ID} \cdot \prod_{v_i=1} S_i \pmod{N},$$

using the stored S_{ID} and S_i values. Correspondingly, the calculation of X_{ref} in the message authentication unit is made according to the formula

$$X_{ref} = F_{ID} \cdot \prod_{v_i=1} F_i \pmod{N},$$

using the stored F_i values, with the F_{ID} value being included in the certified message transmitted from the message source unit to the message authentication unit for use in the computation of X_{ref}.

It will be appreciated that in the third embodiment,

with S_{ID} included in the computation of Y, the requirement that V contains at least two binary 1 bits is reduced to the requirement that V should be non-zero.

The embodiments described hereinabove may be used for any application where it is desired to authenticate entities or the data emanating from them. An important application, however, is to an intelligent financial transaction token or smart card used in Electronic Funds Transfer at the Point of Service (EFTPOS). For several reasons of cost and security it is perceived that the so called "smart card" provides a highly effective technology for EFTPOS.

A fundamental reason for using smart card technology is to enable a transaction to be completed fully off-line from the card issuer's authorization system with a minimum of risk to the various parties affected.

From a risk analysis point of view, the following areas must be considered

(a) Is the card holder legitimate?

(b) Is the card authentic?

(c) Is the implied value loaded into or dispensed by the card authentic?

(d) Is the transaction claim made by the card acceptor authentic?

Card holder authenticity is generally effected by employing a Personal Identification Number (PIN) which is verified by or with the smart card prior to sensitive operations being initiated. Such PIN may be entered via a keyboard such as the keyboard 56, Fig. 2, or by a keyboard (not shown) integral with the card.

It is commonly perceived that card authenticity needs to be established prior to transferring value to prevent bogus funds being loaded into or dispensed by the card. However, this requirement in essence occurs with many

implementations because it is not possible to authenticate at the point of service the value data exchanged.

Therefore, considering the dispersal of value from a card, provided that the card could itself produce an authentication certificate for the data emanating from it such that the certificate could be tested by any other entity, then card authentication is unnecessary. This has significant consequence for remote card authentication or home banking applications, as the need for a trusted card authentication device at the point of card acceptance is eliminated. This possibility also enables any intermediate entity handling the value message between the card and the entity guaranteeing the funds to test the authenticity of the data in order to undertake settlement actions. In this sense, the potential exists for true electronic currency.

Considering the loading of value, if it can be shown that data emanating from a card is authentic, it must be assumed that only an authentic card could perform the certificate calculation corectly. Therefore, if only an authentic card can correctly dispense funds, then the requirement of preventing the loading of bogus value can be readily met by designing authentic cards such that they will reject an attempted loading of bogus value themselves.

Since the card contains the ability to generate certificates, it could therefore check a certificate as well. This could be done in a fourth embodiment of the invention by calculating a certificate for value load data presented to the card in the same manner as done by the card itself and appending that certificate to the value load data. The card could replicate that operation and compare the result with the presented certificate. The presumption is that only the entity guaranteeing dispensed value could correctly load value so

that it is assumed that this entity knows the secret certificate calculation method.

However, this technique would require the entity generating the load value certificate to have available a record of each card's secrets (given the potential size of card networks, the possibility that several value generators may wish to load value, and the highly desirable need to uniquely authenticate each card) this requirement could become impractical.

The primary advantage of the embodiments described hereinabove is that any entity may easily test the authenticity of data emanating from another entity. If it was considered that the source of the value load data was a similar entity to the load accepting entity, then any other entity including the destination card itself could similarly easily test the load data for authenticity prior to acceptance.

Thus, the need to authenticate a card or, conversely, the need for the card to authenticate the load device is eliminated if the techniques of public message authentication as described in the third embodiment are employed.

Thus, the fourth embodiment of the invention provides a means and method for eliminating the need for trusted terminal devices, which may have the capability of adding information or value to the entities in the set, by delivering such information with an authentication certificate such that the member entity can authenticate that information as emanating from the identified source prior to its acceptance. In the fourth embodiment the member entity (smart card) possesses both the ability to generate its own certificates and also test certificates from other entities by employing in the first case the techniques of the third embodiment to generate certi-

ficates and in the second case the complementary techniques of the third embodiment to test certificates.

In this fourth embodiment, the card may additionally contain stored therein the F_i, N and e values appropriate for each value load generator which is authorized by the card issuer to perform the value load function. For convenience, all generators should employ the same public factors F_i and public key e, with individual integrity being obtained by the use of different N values.

Although in the preferred embodiments, the calculations within the card 30, and acceptor device 32, message source unit 30A and message authentication unit 32A have been described as being effected by microprocessors 36, 50, 36A, 50A, it should be understood that in a modification, each microprocessor may be associated with a respective dedicated calculation unit which performs the function

$$f(P) = P.M \pmod{N}.$$

Such dedicated circuitry may use shift register and serial adder/subtractor elements such that a value M is multiplied by a value P while simultaneously the value N is subtracted, if necessary, to yield within a single computation cycle the desired product value P.M (mod N). By this means, the function

$$Y = \prod_{v_i=1} S_i \pmod{N}$$

may be computed with the values S_i being progressively presented as indicated by the values of the bits v_i of V.

The embodiments described above provide a high degree of security both for the authentication of entities and for the certification of messages. However, it should be understood that, depending on system implementation, a sophisticated attacker could compromise a system employing such authentication and/or certification techniques, as will now be explained. Thus, since the factors F_i and S_i are selected for multiplication according to the value of V, it follows that, if the system design permitted an appropriately manipulated authentication device to generate any desired values of V, for example, if the values

$$V_a = 3 \text{ (decimal)} = 011 \text{ (binary)}$$

and $V_b = 7$ (decimal) = 111 (binary)
could be freely chosen, then corresponding Y values

$$Y_a = S_1.S_2 \pmod N$$

and $Y_b = S_1.S_2.S_3 \pmod N$
would be produced.

Since

$$S_3 = Y_b/Y_a = (S_1.S_2.S_3)/(S_1.S_2) \pmod N,$$

S_3 is disclosed. Similarly, any desired S_i can be ascertained, provided that division operations can be effected. Due to the modulus N operation on Y_a and Y_b, simple division will not necessarily yield a correct value. However, since N is a composite of large prime numbers (usually two), then most numbers in the range 1 to N-1 will have a modulo N reciprocal, i.e. given Y, there is, generally, a value Y^{-1}, such that

$$Y.Y^{-1} = 1 \pmod N$$

Known mathematical techniques can be utilized to find such reciprocal value Y^{-1}.
Hence, $S_3 = Y_b.Y_a^{-1} \pmod N$
can be determined, and, by similar techniques, the remaining S_i can also generally be ascertained. Having ascertained the S_i values, the sophisticated attacker,

using suitable hardware could fraudulently effect authentication and certification procedures.

To avoid such an attack, it should be made infeasible to select V values which yield a set of Y values which can be manipulated to yield single factors S_i.

In a fifth embodiment of the invention, this problem is alleviated by including an additional public parity factor F_p and associated secret factor S_p in the system, where

$$S_p = F_p^d \pmod N,$$

and arranging that all Y values are the product of an even number of factors, utilizing S_p if necessary, thus preventing the ascertainment of any single factor. For example, with this arrangement,

for V = 1 (decimal), $Y = S_1.S_p \pmod N$

for V = 2 (decimal), $Y = S_2.S_p \pmod N$

for V = 3 (decimal), $Y = S_1.S_2 \pmod N$, etc.

Thus, in the arrangement described with reference to Fig. 1, a card issuer selects an additional public factor F_p, calculate S_p and store S_p in the cards to be issued. Similarly, in the message certification system described with reference to Figs. 3 and 4, the additional secret parity factor S_p is stored in the PROM 42A and the corresponding public parity factor F_p stored in the RAM 52A. Again, with the unique identification arrangement described with reference to Fig. 5, the secret parity factor S_p is stored in the secure PROM 42B, in addition to the S_{ID} value, and with this arrangement, there is the further advantage that V can be in the full range of 0 to 2^n-1. This is desirable for message certification since it eliminates any need to adjust the message hash result. Thus, with this arrangement,

for V = 0 (decimal), $Y = S_{ID}.S_p \pmod N$

for V = 1 (decimal), $Y = S_{ID}.S_1 \pmod N$

for V = 2 (decimal), $Y = S_{ID}.S_2 \pmod N$

for V = 3 (decimal), $Y = S_{ID}.S_1.S_2.S_p \pmod N$, etc.

Although it could be argued that if the fifth embodiment is utilized, an attacker could selectively extract all factor pairs,

e.g. $S_1.S_2 = V_3.V_0^{-1}$,

and use these pairs to produce bogus certificates in a message certification scheme, such an attack may be infeasible due to the number of pairs needed to be obtained and fraudulently used in systems where n has a suitably large value.

Another way to prevent selective extraction of S_i values by an attacker is to ensure that any Y value is not consistently related to any other Y value. This can be achieved by including a variable component in the Y calculation which cannot be controlled or predicted by an attacker. Such variable component should be chosen from a large enough set of possible component values to make the reoccurrence of any specific value statistically improbable. That is, the number of Y values needing to be obtained to ensure that the same variable component is included in the calculation, should be infeasibly large for an attacker.

Firstly, it will be appreciated that the Y values are in fact a base set of 2^n values pseudo-randomly distributed within the set bounded by 1 and N-1. Secondly, it will be appreciated that the numerical separation of these Y values is in fact precisely determined. Application of an offset value which was applied to all Y values in the base set would in effect produce another set of

precisely separated Y values within the set 1, N-1. Thus, provided that the number of Y sets which could be produced by offset was large enough to be statistically unique, then mathematical extraction of the factors making up a certain Y value would be infeasible, unless the set offset value was known, since the number of valid Y values within the set 1, N-1 would be increased from 2^n to 2^n times the number of Y sets.

In the extreme case, consider that the number of Y sets was N-1 then the number of valid Y values would be $2^n.(N-1)$. This would raise the probability that an entity producing a Y value was authentic, or that a message from the entity was authentic, from 2^n to $2^n.(N-1)$. For typical N values $2^{512} < N < 2^{513}$ then the order of probability of authenticity would be $2n.2^{512}$. This is not true in practice since the total of Y values available is N-1, limiting the probability to 1:(N-1). Clearly since this order of probability far exceeds any reasonable requirement, the number of Y sets could be substantially reduced. If s equals the number of binary bits available to denote the set number then the number of sets would be 2^s giving an authenticity probability of $2^n.2^s$ or 2^{n+s}. Note that in principle n and s could be varied in size to obtain the order of probable authenticity protection desired in the system. However, since the 2^n component may be selectable via V by an attacker the 2^s component should be large enough to make such an attack infeasible. Also, note that n determines the range of V and should be large enough to preclude undetected manipulation of message contents when V results from a hash function of a message.

In such a system it is necessary to communicate to the authenticator the Y set employed for a particular Y calculation by the certifying entity. If this was directly disclosed as an offset value, then the

aforementioned attacks could still be executed since reversing the offset process would yield the original base set of Y values and thus by extraction, the base set of S_i values. Consequently, the offset value or set identifier should be provided in a manner usable by the authenticator for Y testing but not for Y factoring.

For example, it is possible to include in the authentication protocol a value F_{set} which is passed to the authenticator for each Y calculation. F_{set} is produced by the certifier selecting a set number S_{set} and computing

$$F_{set} = S_{set}^{e} \pmod{N}$$

Note that S_{set} cannot be determined from F_{set} without knowledge of d. Thus, for entity authentication, the entity:

(i) Selects an S_{set}

(ii) Computes $F_{set} = S_{set}^{e} \pmod{N}$

(iii) Communicates F_{set} to the authentication device, which

(iv) Selects a V value and communicates this value to the entity, which computes

(v) $Y = S_{set} \cdot \prod_{V_i=1} S_i \pmod{N}$ which it communicates to the authentication device, which tests Y by

(vi) $X_{ref} = F_{set} \cdot \prod_{V_i=1} F_i \pmod{N}$

$= X_{act} = Y^{e} \pmod{N}$.

Note that, since F_{set} is a pseudo-random distribution within the set 1, N-1 from which it is not feasible to determine S_{set}, then it is not necessary to choose S_{set} randomly. The protection from analytical attacks can be obtained merely by ensuring that S_{set} does not predictably repeat within an attack session. One such method to achieve this is to run an incremental count of Y

calculations and to use this count value to update S_{set}. This method has the further advantage of providing to the entity originator a method of cryptographically checking for lost or duplicated messages delivered to him from the source entity.

Thus, in a sixth embodiment of the invention, for message certification,

$$Y = S_{ID} \cdot S_{set} \cdot \prod_{V_i=1} S_i \cdot S_p \pmod{N}$$

where S_{set} = a function of the counter value

$S_{ID} = F_{ID}^{d} \pmod{N}$

$S_i = F_i^{d} \pmod{N}$

$S_p = F_p^{d} \pmod{N}$ optionally included if V has even parity,

and the certificate Y is calculated across a message including F_{ID}, F_{set} therein, where $F_{set} = S_{set}^{e} \pmod{N}$.

To generate the S_{set} counter values a hardware counter could be provided in a smart card or entity to be authenticated, such as the card 30, Fig. 2, or in a message source unit such as the message source unit 30A, Fig. 3. Alternatively, the microprocessor 36 or 36A therein could be programed to provide a counting operation using storage locations in the RAM memories 38 or 38A. An analogous arrangement could be utilized when a unique identifier factor S_{ID} and associated F_{ID} are employed as described hereinabove with reference to the third embodiment of the invention.

In the just mentioned system the protocol is enlarged by the inclusion of F_{set}. This is unimportant for interactive entity authentication by locally communicating devices but may be an unacceptable overhead for message certification.

A further method of pseudo-randomly varying the base set of Y values which does not add significantly to the protocol is to utilize precalculated offset values the selection of which is advised to the authentication device.

In a seventh embodiment of the invention, V, which is made up of n bits, is split into two parts, V_s and V_a, where V_s is chosen by the certifier, and V_a as before is chosen in the authentication device (or determined by the message content). The number of bits in each of V_s and V_a is predetermined. For example, where n=32, each of V_s and V_a could have 16 bits. The bits of V_s are used to select the S_{set} offset value with the bits of V_a being used to select the S_a values. Note also that the S_{set} offset values can be combined to yield 2^{ns} offset values, where ns is the number of base offset values available.

Thus, in the seventh embodiment,

$$Y = \prod_{V_{si}=1} S_{si} \cdot \prod_{V_{ai}=1} S_{ai} \pmod N;$$

$$X_{ref} = \prod_{V_{si}=1} F_{si} \cdot \prod_{V_{ai}=1} F_{ai} \pmod N; \text{ and}$$

$X_{act} = Y^e \pmod N$, as before.

The values $S_{si} = F_{si}^d \pmod N$ are stored by the certifier (smart card or message source unit) and used in a similar manner to the S_{ai} values, but selected by the certifier pseudo-randomly.

The values F_{si} are made publicly available in the same manner as the F_{ai} values.

In this embodiment, V_s rather than F_{set} would be

included (and hashed for V_a) in the certified message.

Thus, for message certification where the unique identifier factors

S_{ID} and F_{ID} are utilized,

$M = V_S, F_{ID}$, Message.

As in the second embodiment, a change-sensitive transformation H of the aggregate message M is formed, and the value of V_a derived therefrom. The following calculations are then effected:

$$Y = S_{ID} \cdot \prod_{V_{si}=1} S_{si} \cdot \prod_{V_{ai}=1} S_{ai} \pmod N; \text{ and}$$

$$X_{ref} = F_{ID} \cdot \prod_{V_{si}=1} F_s \cdot \prod_{V_{ai}=1} F_{ai} \pmod N.$$

It can be seen from the above that the authenticity of a particular Y value is as before 1:N. The authenticity of the entity producing the Y value (entity forgery) is determined by the number of bits in V_S and V_a and is therefore $1:2^{ns+na}$.

CLAIMS

1. A method of manufacturing an entity (30, 30A), including the steps of:

(a) selecting a modulus N which is a product of at least two prime numbers;

(b) selecting an integer e which is relatively prime to $\varphi(N)$, where $\varphi(N)$ is Euler's totient function of N; and

(c) determining an integer d such that $e.d = 1 \pmod{\varphi(N)}$, characterized by the steps of:

(d) selecting a set of n public factors $F_1, \ldots, F_n$ $(0<F_i<N)$;

(e) calculating $S_i = F_i^d \pmod{N}$ for i=1, ..., n; and

(f) storing the n values S_i (i=1, ..., n) and the value N in said entity.

2. A method according to claim 1, characterized in that said n values S_i are stored in a programmable read-only memory (PROM) (42, 42A, 42B) included in said entity (30, 30A).

3. A method according to claim 2, characterized in that said entity (30, 30A) includes processing means (36, 36A) and input/output means (44, 44A).

4. A method according to claim 1, characterized by the steps of:

(g) assigning a public factor F_{ID} unique to said entity;

(h) computing $S_{ID} = F_{ID}^d \pmod{N}$; and

(i) storing the value S_{ID} in said entity.

5. A method of authenticating an entity (30, 30A) according to any one of claims 1 to 4, characterized by the steps of:

(j) placing said entity (30, 30A) in communication with an authentication device (32, 32A);

(k) generating in said authentication device (32, 32A) an n-bit binary string $V = v_i$ (i=1, ..., n)

(l) transmitting said binary string V to said entity (30, 30A);

(m) calculating, in said entity (30, 30A)

$$Y = \prod_{v_i=1} S_i \pmod N;$$

(n) transmitting Y to said authentication device (32, 32A);

(o) calculating, in said authentication device (32, 32A)

$$X_{ref} = \prod_{v_i=1} F_i \pmod N; \text{ and}$$

$$X_{act} = Y^e \pmod N; \text{ and}$$

(p) comparing X_{ref} and X_{act}.

6. A method according to claim 5, characterized in that said authentication device (32, 32A) includes storage means (52, 52A) adapted to store said public factors F_1, ..., F_n, and the values of N and e.

7. A method according to claim 6, characterized by the step of repeating said steps (k) to (p) a plurality of times, using random values of V.

8. A method of certifying a message M generated by or presented to an entity (30, 30A) manufactured according to any one of claims 1 to 4 characterized by the steps of:

(q) computing a change-sensitve transformation H of said message M;

(r) generating an n-bit binary string $V = v_i$ (i=1, ..., n), using the computed value of H;

(s) computing

$$Y = \prod_{v_i=1} S_i \pmod N; \text{ and}$$

(t) appending Y as a message authentication code (MAC) certificate to said message M.

9. A method according to claim 8, characterized in that said step of (q) computing said change-sensitive transformation H is effected by computing

$H = M^e \pmod N$.

10. A method according to claim 8 or 9, characterized in that said step of (r) generating an n-bit binary string V is effected by the steps of:

(u) converting H to a binary value J;

(v) segmenting J into sub-fields of length n; and

(w) adding together the individual sub-fields modulo 2 to form said n-bit binary string V.

11. A method according to any one of claims 5 to 10, characterized in that, in said step (m) and said step (s) the value of Y is calculated according to the formula

$$Y = S_{set} \cdot \prod_{V_i=1} S_i \pmod N,$$

where S_{set} is selected in said entity (30,30A); by the steps of

(x) computing in said entity (30,30A) $F_{set} = S_{set}^e \pmod N$, and

(y) transmitting F_{set} to said authentication device (32,32A);

and in that in said step (o), the value of X_{ref} is calculated according to the formula

$$X_{ref} = F_{set} \cdot \prod_{V_i=1} F_i \pmod N.$$

12. A method according to claim 11, characterized in that S_{set} is selected in accordance with a count value which is incremented for each Y caluculation.

13. A method according to claim 12, characterized in that S_{set} is determined by computing,

in said entity (30,30A), a product including a selection of a set S_{si} of said factors S_i, said selection being in accordance with a binary string $V_s = v_{si}$ generated in said entity (30,30A), whereby the value of Y is calculated according to the formula.

$$Y = \prod_{v_{si}=1} S_{si} \cdot \prod_{v_{ai}=1} S_{ai} \pmod N,$$

wherein the v_{ai} values corresponding to the bits of said n-bit binary string generated in said authentication device (32,32A);

by the step of:

(z) transmitting V_s to said authentication device (31,32A), and in that the value of X_{ref} is calculated in said authentication device (32,32A) according to the formula

$$X_{ref} = \prod_{v_{si}=1} F_{si} \cdot \prod_{v_{ai}=1} F_{ai} \pmod N.$$

14. A method according to any one of claims 5 to 13, characterized in that, in said step (m) and said step (s), the value of Y is calculated utilizing selectively an additional predetermined factor S_p, such that the total number of factors included in the calculation of Y is even, and in that, in said step (o), the value of X_{ref} is correspondingly calculated, utilizing selectively an additional factor F_p, where $S_p = F_p^d$.

15. An entity (30,30A), including processing means (36,36A), input/output means (44,44A) and memory means (42,42A,42B), characterized in that said memory means (42,42A,42B) has stored therein a modulus N which is the product of at least two prime numbers and a set of n factors S_i (i=1, ..., n) where

$$S_i = F_i^d \pmod N,$$

where d is the secret key counterpart of a public key e, associated with the modulus N, and F_i (i=1, ..., n) are

n public factors, $0<F_i<N$, and in that said processing means (36,36A) is adapted to compute

$$Y = \prod_{V_i=1} S_i \pmod N$$

where $V = V_i$ is an n-bit binary string.

16. An entity according to claim 15, characterized in that said memory means (42A,42B) is further adapted to store the value of said public key e and in that said processing means is further adapted to compute

$$H = M^e \pmod N$$

where M is a message to be transmitted by said entity (30,30A), to convert H to a binary n-bit vector V, and to compute

$$Y = \prod_{v_i=1} S_i \pmod N.$$

using the bits v_i of the computed vector V, and in that said input/output means (44,44A) is adapted to transmit Y as a message authentication code (MAC) associated with said message.

17. An entity according to claim 15 or 16, characterized in that said memory means (42B) has stored therein a public factor F_{ID} unique to said entity, and a value S_{ID}, where

$$S_{ID} = F_{ID}^{d} \pmod N.$$

18. An entity according to any one of claims 15 to 17, characterized in that the value of Y includes an additional factor S_{set} which is dependent on a count value which is incremented for each Y calculation.

19. An entity according to any one of claims 15 to 18, characterized in that said memory means (42,42A,42B) has stored therein an additional parity factor S_p, and in that said processing means (36,36A) is

adapted to compute the value of Y by selectively including said additional parity factor S_p in the expression for Y, such that the total members of factors included in the calculation of Y is even.

20. An authentication device (32, 32A) for use with an entity (30, 30A) according to any one of claims 15 to 19, including further processing means (50, 50A), further input/output means (64, 64A) and further memory means (52, 52A), characterized in that said further memory means (52, 52A) has stored therein said n public factors F_i (i=1, ..., n), said modulus N, and said public key e, and wherein said further processing means (50, 50A) is adapted to compute

$$X_{ref} = \prod_{v_i=1} F_i \pmod N; \text{ and}$$

$$X_{act} = Y^e \pmod N$$

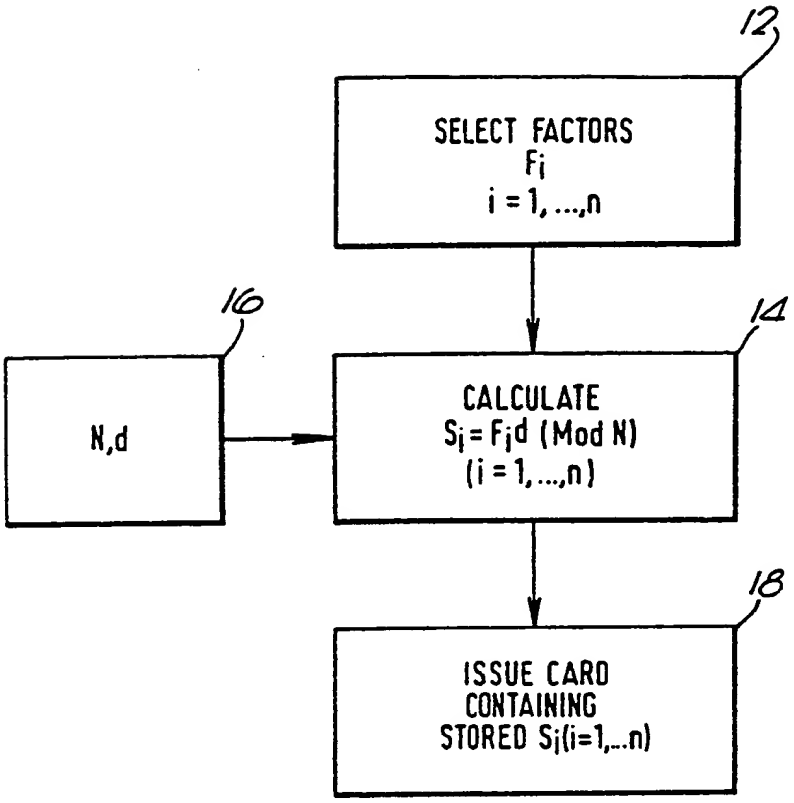
using the stored values of F_i, N and e, and to compare X_{ref} with X_{act}.

21. An authentication device according to claim 20 for use with an entity according to claim 17, characterized in that said further processing means is further adapted to compute

$$X_{ref} = F_{ID} \cdot \prod_{v_i=1} F_i \pmod N$$

22. An entity according to any one of claims 15 to 19, characterized in that said entity incorporates an authentication device according to claim 20 or claim 21.

FIG. 1


12
SELECT FACTORS
Fi
i = 1, ...,n
16
N,d
14
CALCULATE
Si = Fid (Mod N)
(i = 1, ...,n)
18
ISSUE CARD
CONTAINING
STORED Si(i=1,...n)

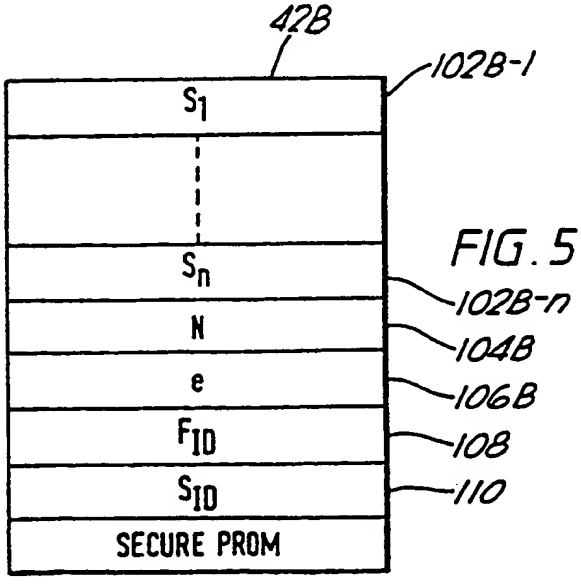

42B
102B-1
S1
Sn
102B-n
N
104B
e
106B
FID
108
SID
110
SECURE PROM

FIG. 5

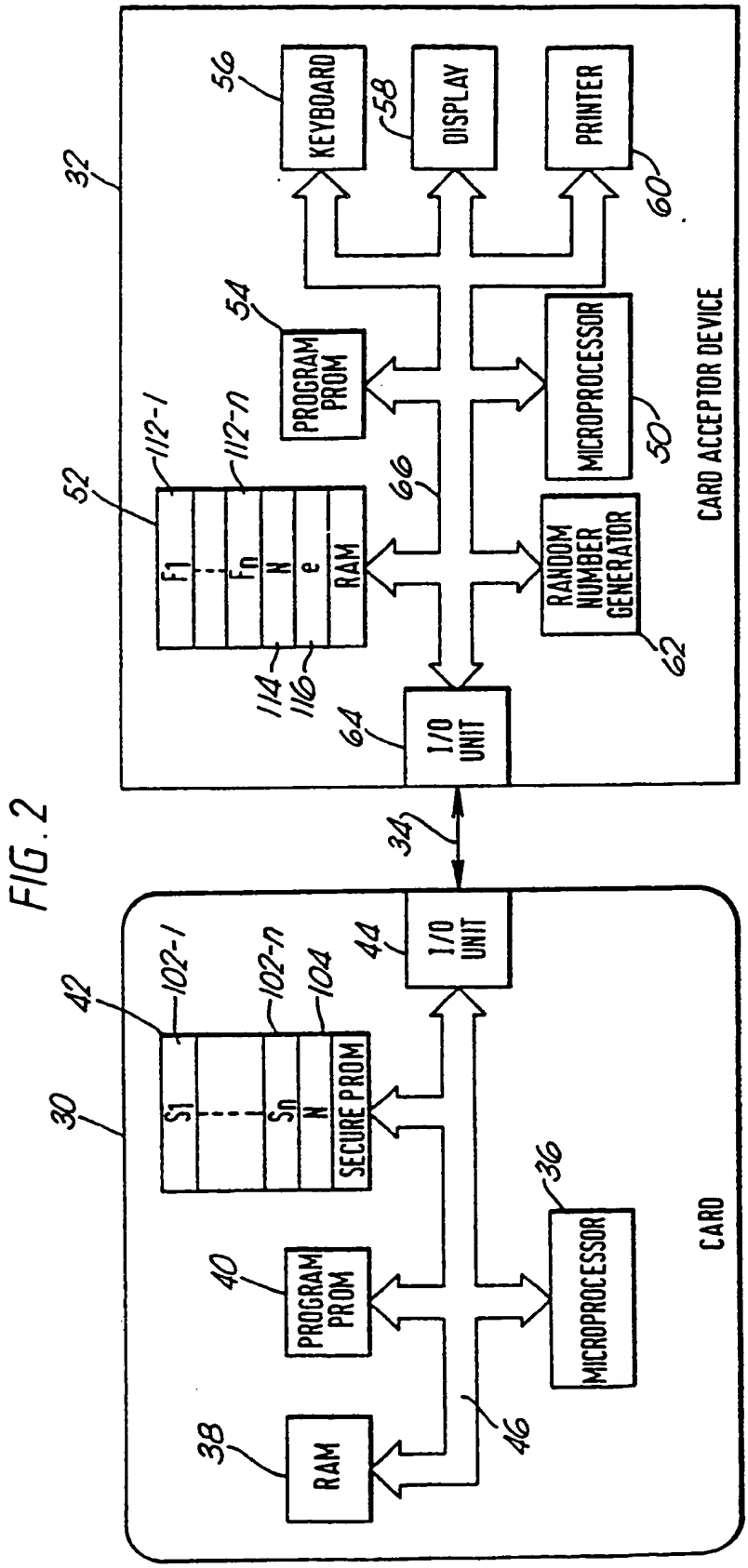

CARD ACCEPTOR DEVICE
32
KEYBOARD
56
DISPLAY
58
PRINTER
60
PROGRAM PROM
54
MICROPROCESSOR
50
52
112-1
112-n
F1
Fn
N
e
RAM
66
RANDOM NUMBER GENERATOR
62
114
116
I/O UNIT
64
34
CARD
30
I/O UNIT
44
42
102-1
102-n
104
S1
Sn
N
SECURE PROM
36
PROGRAM PROM
40
MICROPROCESSOR
RAM
38
46

FIG. 2

FIG. 3

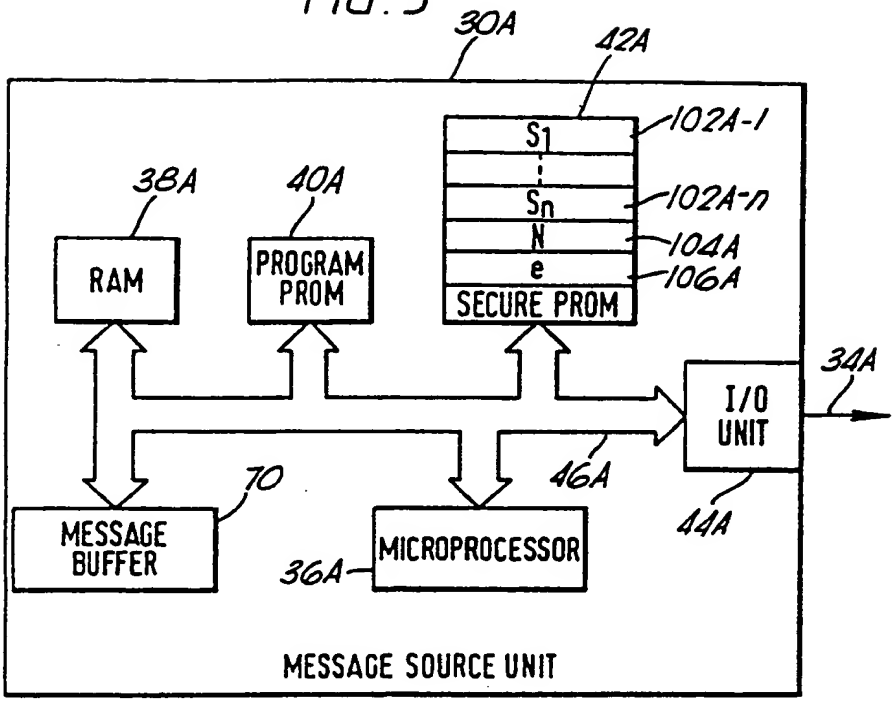
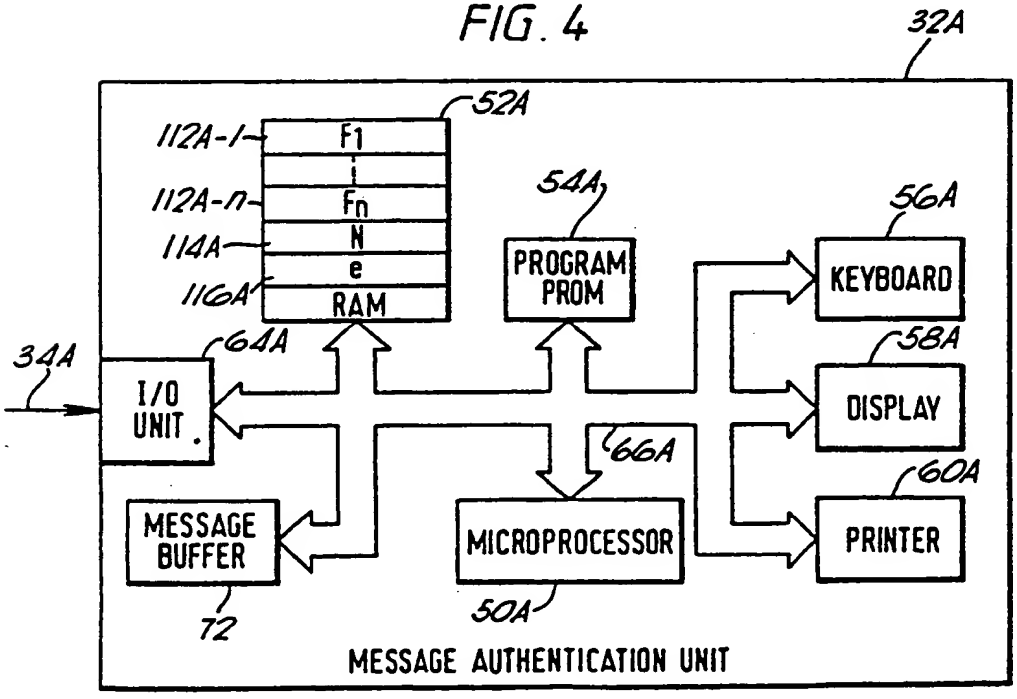

30A
42A
S1
Sn
N
e
SECURE PROM
102A-1
102A-n
104A
106A
38A
40A
RAM
PROGRAM PROM
34A
I/O UNIT
46A
44A
70
MESSAGE BUFFER
36A
MICROPROCESSOR
MESSAGE SOURCE UNIT

FIG. 4


32A
52A
112A-1
F1
112A-n
Fn
114A
N
e
116A
RAM
54A
PROGRAM PROM
56A
KEYBOARD
34A
64A
I/O UNIT
58A
DISPLAY
66A
60A
MESSAGE BUFFER
MICROPROCESSOR
PRINTER
72
50A
MESSAGE AUTHENTICATION UNIT

INTERNATIONAL SEARCH REPORT

International Application No PCT/US 89/01944

I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all) [6]

According to International Patent Classification (IPC) or to both National Classification and IPC

IPC[4]: G 07 F 7/10, H 04 L 9/00

II. FIELDS SEARCHED

Minimum Documentation Searched [7]

Classification System	Classification Symbols
IPC[4]	G 07 F, H 04 L

Documentation Searched other than Minimum Documentation to the Extent that such Documents are included in the Fields Searched [8]

III. DOCUMENTS CONSIDERED TO BE RELEVANT [9]

Category [10]	Citation of Document, [11] with indication, where appropriate, of the relevant passages [12]	Relevant to Claim No. [13]
Y	US, A, 4549075 (SAADA et al.) 22 October 1985, see abstract; figure 1; column 2, line 59 - column 4, line 42; column 5, line 11 - column 7, line 14; column 8, line 1 - column 10,line 19; claims 1-7	1-21
Y	US, A, 4405829 (RIVEST et al.) 20 September 1983, see the whole document	1-21
A	US, A, 4351982 (MILLER et al.) 28 September 1982, see abstract; figures 1,2; column 2, line 3 - column 6, line 26; column 7, line 52 - column 10, line 39; claims 1-12,21	1-21
A	US, A, 4349695 (MORGAN et al.) 14 September 1982, see abstract; figures 2-4; column 2, lines 21-32; column 4, line 31 - column 8, line 48; claims 1,2	1-3,5-11, 13-16,20-22

./.

* Special categories of cited documents: [10]

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

IV. CERTIFICATION

Date of the Actual Completion of the International Search	Date of Mailing of this International Search Report
31st August 1989	02 OCT. 1989
International Searching Authority	**Signature of Authorized Officer**
EUROPEAN PATENT OFFICE	T.K. WILLIS

III. DOCUMENTS CONSIDERED TO BE RELEVANT (CONTINUED FROM THE SECOND SHEET)

Category *	Citation of Document, with indication, where appropriate, of the relevant passages	Relevant to Claim No
A	US, A, 4679236 (DAVIES) 7 July 1987, see abstract; figure 6; column 1, line 63 - column 3, line 66; column 6, line 27 - column 7, line 30; claims 1-19	1-3,5-11, 13-16, 20-22
A	EP, A, 0218305 (CHAUM) 15 April 1987, see abstract; column 4, line 1 - column 15, line 26; claims 1-6	1-3,5-8, 13-15,20.
A	US, A, 4723284 (MUNCK et al.) 2 February 1988, see the whole document	1,2,5,6, 13,20
A	US, A, 4408203 (CAMPBELL) 4 October 1983, see abstract; claims 1-8	1
P,A	US, A, 4797920 (STEIN) 10 January 1989, see the whole document	1,5,8,15
A	17th ACM Symposium on Theory of Computing, May 1985, ACM S. Goldwasser et al.: "The knowledge complexity of interactive proof-systems", pages 291-304, see paragraphs 4.2; pages 298-300	1,5,8,15
P,A	ACM Transactions on Computer Systems, vol. 6, no. 4, November 1988 ACM (New York, NY, US) T. Okamoto: " A digital multisignature scheme using bijetive public-key cryptosystems", pages 432-441, see the whole article	1
A	Computers & Security, vol. 5, no. 3, September 1986 Elsevier Science Publishers (North-Holland) (Amsterdam, NL) G.J.M. Pluimakers et al.: "Authentication: a concise survey", pages 243-250, see the whole article	1

ANNEX TO THE INTERNATIONAL SEARCH REPORT ON INTERNATIONAL PATENT APPLICATION NO. US 8901944 SA 28614

This annex lists the patent family members relating to the patent documents cited in the above-mentioned international search report. The members are as contained in the European Patent Office EDP file on 22/09/89
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
US-A- 4549075	22-10-85	FR-A,B	2530053	13-01-84
		DE-A-	3374039	12-11-87
		EP-A,B	0100260	08-02-84
		JP-A-	59062241	09-04-84
US-A- 4405829	20-09-83	None		
US-A- 4351982	28-09-82	AU-B-	544169	16-05-85
		AU-A-	8006982	01-07-82
		BE-A-	891490	31-03-82
		CA-A-	1173538	28-08-84
		CH-B-	660822	15-06-87
		FR-A,B	2496303	18-06-82
		GB-A,B	2101855	19-01-83
		NL-T-	8120500	01-10-82
		SE-A-	8204697	13-08-82
		WO-A-	8202129	24-06-82
US-A- 4349695	14-09-82	None		
US-A- 4679236	07-07-87	None		
EP-A- 0218305	15-04-87	US-A-	4759064	19-07-88
US-A- 4723284	02-02-88	None		
US-A- 4408203	04-10-83	US-A-	4259720	31-03-81
US-A- 4797920	10-01-89	None		